Short Read Workshop Day 6
Introduction to R and RStudio

Rutendo Sigauke
2024

Day 6 Overview

1. Running `R` in the terminal
2. Running `R` in `RStudio`
3. Submitting `R script` as an sbatch job

Goal of the day

Learn how to run R code!

Practice installing packages, tidying data, saving files and plotting.

What is R?

- R is a free statistical computing and graphing software
- Can be installed from their website https://www.r-project.org/
- R can be run in a few environments:
  - RStudio
  - Jupyter

There are different ways to interact with R

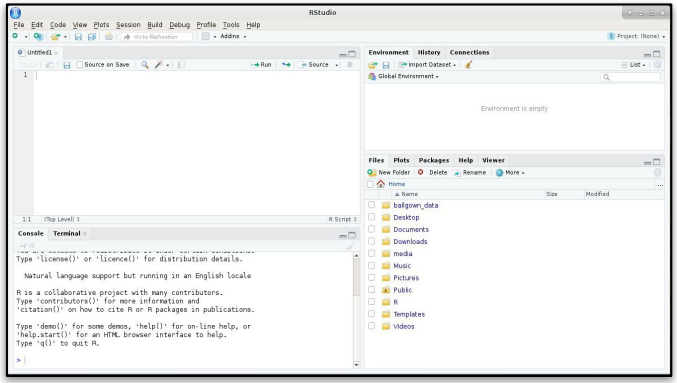
R console

```
(base) cu-biot-14-10:~ rutendo$ R

R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

>
```

Enter **R code** here

Interactive

R Studio



Enter **R code** and **visualize plots**

More interactive

Submit an R script as a job

```
#!/bin/bash
#
#SBATCH --job-name=feature_counts                    # Job name
#SBATCH --mail-type=ALL                  # Mail events (NONE, BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=email@colorado.edu   # Where to send mail
#SBATCH --nodes=1                                    # Number of cores job will run on
#SBATCH --ntasks=4                                   # Number of CPU (processers, tasks)
#SBATCH --time=1:00:00                               # Time limit hrs:min:sec
#SBATCH --partition compute                          # Job queue
#SBATCH --mem=4gb                                    # Memory limit
#SBATCH --output=/scratch/Users/rutendos/e_and_o/%x_%j.out
#SBATCH --error=/scratch/Users/rutendos/e_and_o/%x_%j.err

################## SET VARIABLES #####################################

FEATURECOUNTS=/scratch/Users/rutendos/day6/featureCounts/scripts/d6_featureCounts.R

######################################################################
################ PRINT JOB INFO ######################################

printf "Sample ID: $ROOTNAME"
printf "\nDirectory: $PROJECT"
printf "\nRun on: $(hostname)"
printf "\nRun from: $(pwd)"
printf "\nScript: $0\n"
date

printf "\nYou've requested $SLURM_CPUS_ON_NODE core(s).\n"

######################################################################

Rscript $FEATURECOUNTS
```

Run **R script** here

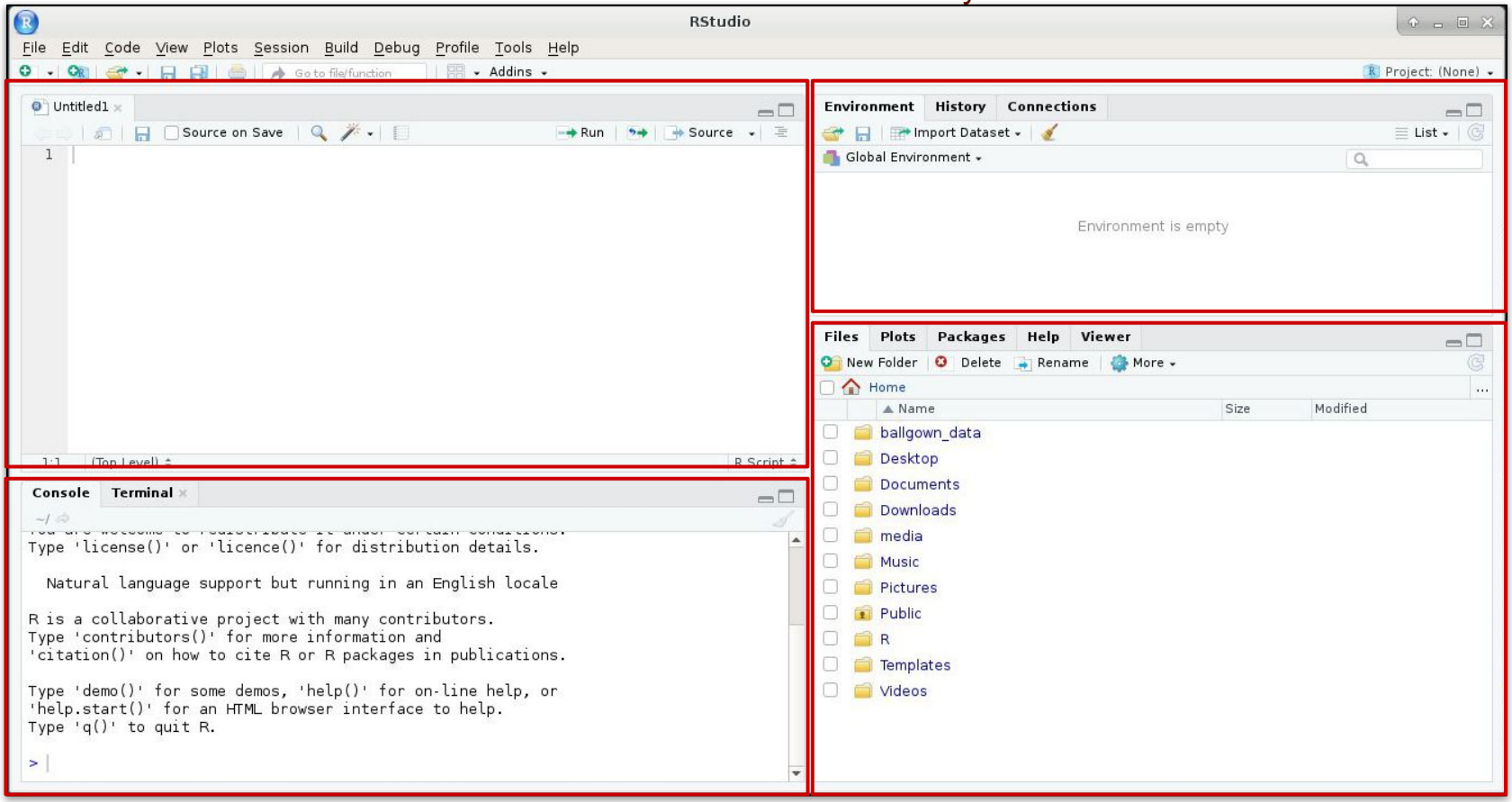
Least interactive

For more compute intensive scripts

Summary of RStudio

R scripts, R markdown, R notebooks

Summary of all the data loaded in Rstudio



R console, Terminal

Directories, Plots, Packages...

Brief introduction to R syntax

Getting Help

Accessing the help files

?mean
Get help of a particular function.
help.search('weighted mean')
Search the help files for a word or phrase.
help(package = 'dplyr')
Find help for a package.

More about an object

str(iris)
Get a summary of an object's structure.
class(iris)
Find the class an object belongs to.

Using Libraries

install.packages('dplyr')
Download and install a package from CRAN.

library(dplyr)
Load the package into the session, making all its functions available to use.

dplyr::select
Use a particular function from a package.

data(iris)
Load a built-in dataset into the environment.

Working Directory

getwd()
Find the current working directory (where inputs are found and outputs are sent).

setwd('C://file/path')
Change the current working directory.

Use projects in RStudio to set the working directory to the folder you are working in.

Vectors

Creating Vectors

c(2, 4, 6)	2 4 6	Join elements into a vector
2:6	2 3 4 5 6	An integer sequence
seq(2, 3, by=0.5)	2.0 2.5 3.0	A complex sequence
rep(1:2, times=3)	1 2 1 2 1 2	Repeat a vector
rep(1:2, each=3)	1 1 1 2 2 2	Repeat elements of a vector

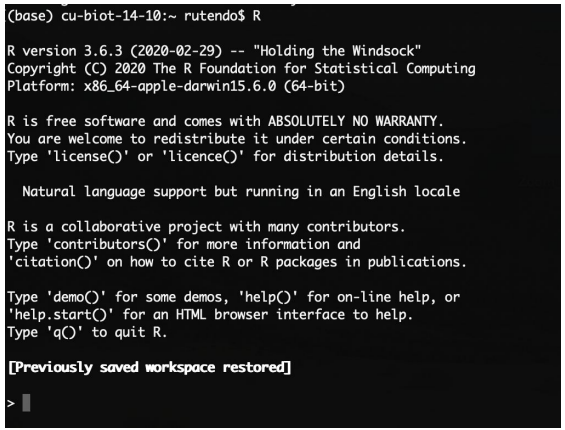
Reading and Writing Data

Input	Ouput	Description
df <- read.table('file.txt')	write.table(df, 'file.txt')	Read and write a delimited text file.
df <- read.csv('file.csv')	write.csv(df, 'file.csv')	Read and write a comma separated value file. This is a special case of read.table/ write.table.
load('file.RData')	save(df, file = 'file.Rdata')	Read and write an R data file, a file type special for R.

https://iqss.github.io/dss-workshops/R/Rintro/base-r-cheat-sheet.pdf

R you ready to learn some R?

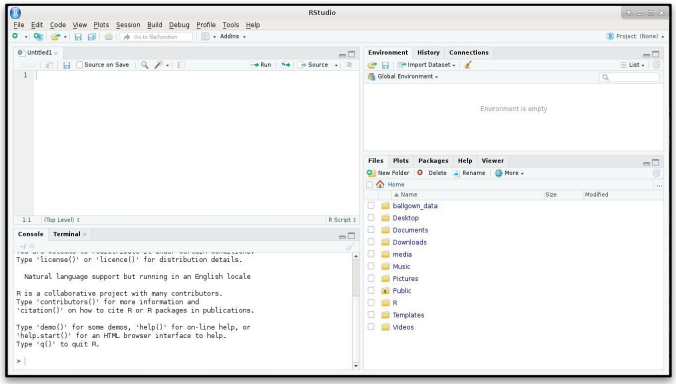
- Let's go over the `Day6_worksheet1_Introduction_to_R.md` worksheet:
  - Introduction to R in the terminal
  - Learn basic R commands



R console

Learning R in RStudio

- In `Day6_worksheet2_R_in_Rstudio.md` (Section A):
- We will go over the `Learning_R.R` worksheet in R Studio:
  - Introduction to R and R Markdown
  - Introduction to the iris dataset
  - Installing and loading libraries
    - tidyverse
  - Generating summary statistic in R
  - Making plots with ggplot2
  - Manipulating data.frames



R Studio

Challenge Question

- How would you perform a computationally intensive R job?
  - i.e. Requires more memory than on your personal computer.

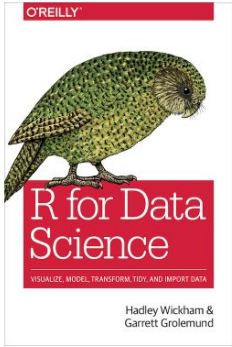
Writing an R script to submit on a supercomputer

- Follow `Day6_worksheet2_R_in_Rstudio.md` (Section B):
- Edit `Learning_R_submit_aws.R`
  - Save plots and tables to a working directory in the script
- Run the R script as a job on AWS
  - Use the `RScript` command to call your script

```
#!/bin/bash

#SBATCH --job-name=feature_counts                          # Job name
#SBATCH --mail-type=ALL                          # Mail events (NONE, BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=email@colorado.edu           # Where to send mail
#SBATCH --nodes=1                                # Number of cores job will run on
#SBATCH --ntasks=4                               # Number of CPU (processers, tasks)
#SBATCH --time=1:00:00                           # Time limit hrs:min:sec
#SBATCH --partition compute                      # Job queue
#SBATCH --mem=4gb                                # Memory limit
#SBATCH --output=/scratch/Users/rutendos/e_and_o/%x_%j.out
#SBATCH --error=/scratch/Users/rutendos/e_and_o/%x_%j.err

################## SET VARIABLES #####################################

FEATURECOUNTS=/scratch/Users/rutendos/day6/featureCounts/scripts/d6_featureCounts.R

######################################################################
################ PRINT JOB INFO ######################################

printf "Sample ID: $ROOTNAME"
printf "\nDirectory: $PROJECT"
printf "\nRun on: $(hostname)"
printf "\nRun from: $(pwd)"
printf "\nScript: $0\n"
date

printf "\nYou've requested $SLURM_CPUS_ON_NODE core(s).\n"

######################################################################

Rscript $FEATURECOUNTS
```

More resources for R

- ggplot2 website https://ggplot2.tidyverse.org/
- R-bloggers https://www.r-bloggers.com/
- Quick-R https://www.statmethods.net/
- R for Data Science (by Hadley Wickham & Garrett Grolemund) http://r4ds.had.co.nz/



Homework

1. Complete the `Learning_R_Additional_Practice.R`

This homework will go over most of the topics covered today, but on a different dataset. There will be more advanced questions that build on what was in the inclass session.

2. For **Project A (single-cell RNA-seq)**

   - Install `Seurat` and `CellChat` (Instructions are on GitHub in the Project A folder)

     Install this on your local machine.

3. For **Project B (bulk RNA-seq)**

   - Install `rsubread`

     Install this in R on AWS.

   - install `DESeq2`

     Install this on your local machine.

This takes a long time, so get this installed before Day7.